NYU Tandon School of Engineering
CS-GY 6083, Principles of Database Systems, Fall 2024

Practice Material #4

<u>**Problem 1**</u>

Suppose we design a database with only one table schema as shown below to model the payroll system for a company

Payroll(eID, dept, position, month, year, branchID, wage, hoursWorked, branchCity, branchState)

We have the following detailed specifications for the database system:
Branches are identified by a **branchID**, besides, the system also records associated branchCity and branchState information. Employees are identified by an **eID**, also necessary information like department the employee is working for (dept), current position in the department and working for which branch (branchID) is recorded. Any employees with the same position have the same wage. Besides, the system also records the hours worked by each employee in each month of each year, and no employee changes department, working branch, or position during the period covered by any payroll table.

1. Write the following SQL queries:
    a. Count the number of employees who work for branches in Brooklyn, NY.
    b. Select the employee and correspondent working hours in March, 2024 with the highest total wage (hoursWorked * wage)
    c. Output the average salary of each department. (You may get the average salary of each employee throughout this employee's career first)

2. Give an example of a trivial functional dependency in this schema. (Recall that a functional dependency alpha $\alpha \rightarrow \beta$ is trivial if $\beta \subseteq \alpha$)
3. List the functional dependencies corresponding to each of the following parts of the data description:
    a. Branches are identified by a **branchID**, besides, the system also records associated branchCity and branchState information
    b. Besides, the system also records the hours worked by each employee in each month of each year.
    c. No employee changes department, working branch, or position during the period covered by any payroll table.
4. Derive a collection F of functional dependencies such that F+ is the closure of the set of functional dependencies implied by the description of the data. F+ should include the functional dependencies in (3) *and any others that are implied by the textual description of the data*.
5. Is your answer to (4) a canonical cover? If not, simplify it to derive a canonical cover.

6. Derive all candidate keys for this table.
7. Show that the Payroll schema is not in BCNF
8. Decompose Payroll into a collection of schemas each of which is in BCNF. Show your work.
9. Illustrate that your decomposition is a lossless join decomposition as follows:
   a. Show a table on Payroll with a few rows of data including some redundancy due to the functional dependencies
   b. Project your table onto the decomposed schemas (showing tables on each of the decomposed schemas)
   c. Join them back together and comment on whether they are the same as the origins table
   [This does not prove that the decomposition is a lossless join, but will demonstrate some understanding of the concept]
10. Is the decomposition in (8) dependency-preserving? Why or why not?
11. Rewrite the queries in (1) using the decomposed database schema in (7).
12. Suppose the company updated its wage system, now the wage will depend on position, the branch city, and the branch state.
    a. Write a functional dependency that characterizes this
    b. Which of the original functional dependencies from (5) no longer hold?
    c. Is your result in (8) still dependency-preserving? Justify your answer.
    d. **Optionally,** decompose Payroll into a collection of schemas that are in 3NF and compare the pros and cons of this decomposition, compared to the one in (8). This is not for extra credit, just for practice with 3NF if you're interested.

Problem 2:

Consider a relational schema R = (A, B, C, D, E, F, G) satisfying the following functional dependencies:
F = {D,F->A; E->A;F,B->C; D->A;E->G;F->B,G}
   a. Derive all candidate keys for this schema.
   b. Derive a canonical cover of the functional dependencies in F.
   c. Is the above schema in BCNF? Prove or disprove. If it is not in BCNF, convert it into BCNF.
   d. Is the BCNF schema from c) dependency-preserving? Prove or disprove. [OPTIONAL If not, convert it into 3NF.]